

Parameter estimation of hidden Markov models: comparison of EM and quasi-Newton methods with a new hybrid algorithm

Sidonie Foulon, Thérèse Truong, Anne-Louise Leutenegger & Hervé Perdry

Sidonie Foulon - MaSeMo workshop - July, 4th, 2025

Plan

1. Introduction
2. Methods
 - a. Algorithms
 - b. Examples / data
3. Results
 - a. Application on the examples
 - b. Convergence points
4. Conclusion & Discussion

Introduction : Hidden Markov models

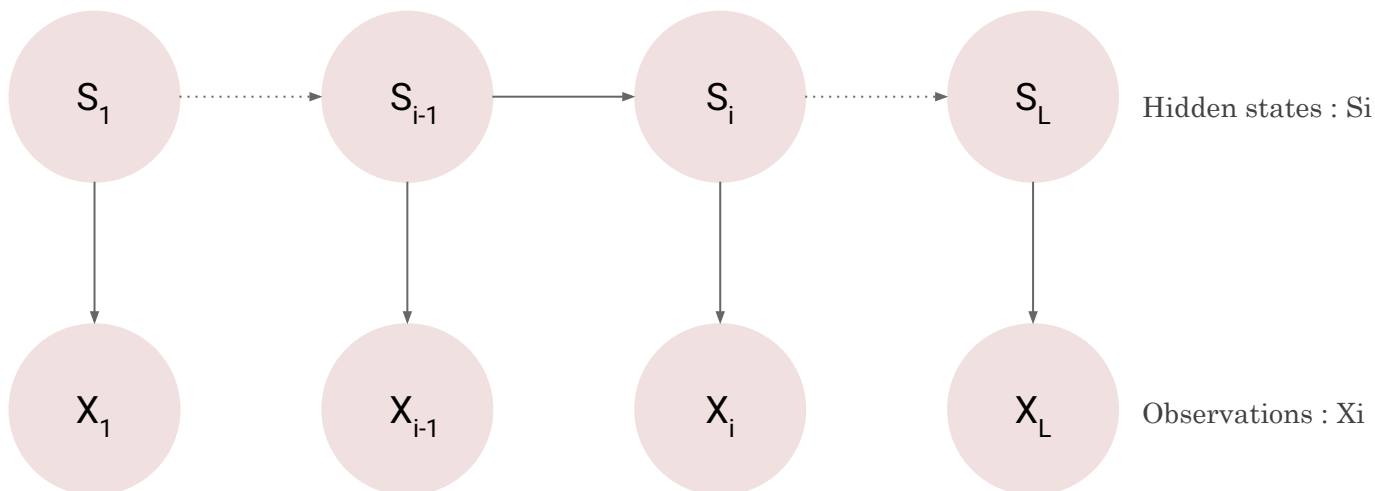
A characteristic of a Markov chain S_i is the homogeneity of the sequence.

⇒ sometimes contradicted by the observations.

To model such situations :

- ❖ add an **observable** layer of variables to the model
- ❖ related to the sequence of **hidden** variables

⇒ **Hidden Markov Models (HMM)**



Introduction : Hidden Markov models

A characteristic of a Markov chain S_i is the homogeneity of the sequence.

⇒ sometimes contradicted by the observations.

To model such situations :

- ❖ add an **observable** layer of variables to the model
- ❖ related to the sequence of **hidden** variables

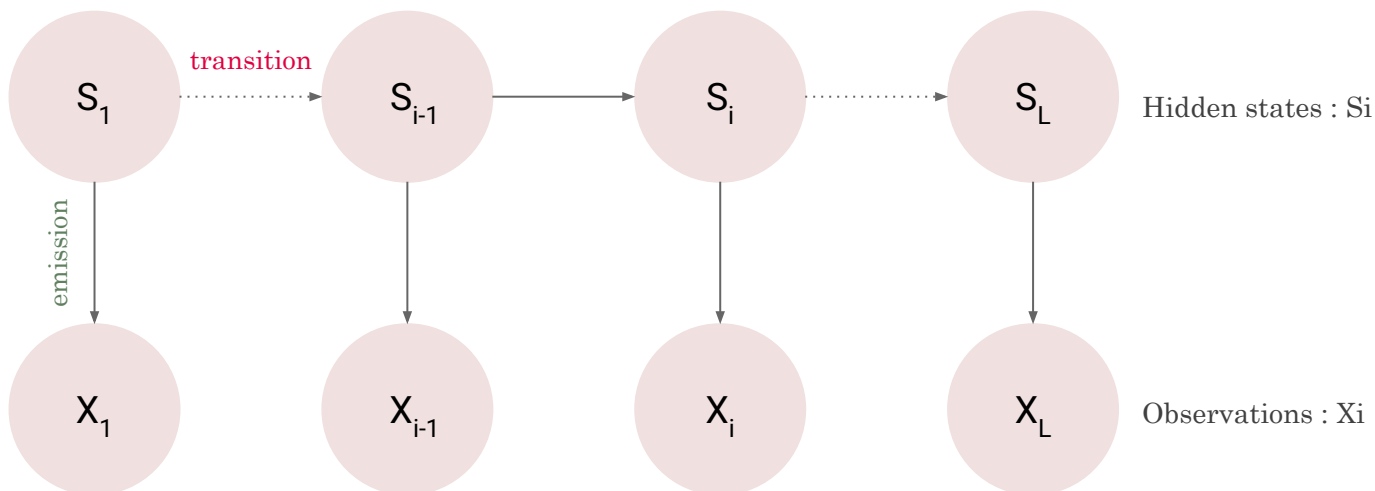
Transition probabilities :

$$T(s, t) = \mathbb{P}(S_i = t | S_{i-1} = s)$$

Emission probabilities :

$$E(x, s) = \mathbb{P}(X_i = x | S_i = s)$$

⇒ **Hidden Markov Models (HMM)**



Aim :

HMM parameters estimation

Introduction : Parameter estimation

Classical optimisation methods to estimate HMM parameters θ

Expectation-Maximisation (EM) algorithm :

Dempster et al., 1977; Baum and Petrie, 1966; Welch, 2003

Estimation of parameters by a variant of EM algorithm : Baum-Welch algorithm.

Direct maximisation of likelihood (DML) :

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

Likelihood maximisation by a quasi-Newton type method (L-BFGS-B).

Introduction : Parameter estimation

Classical optimisation methods to estimate HMM parameters θ

Expectation-Maximisation (EM) algorithm :

Dempster et al., 1977; Baum and Petrie, 1966; Welch, 2003

Estimation of parameters by a variant of EM algorithm : Baum-Welch algorithm.

Direct maximisation of likelihood (DML) :

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

Likelihood maximisation by a quasi-Newton type method (L-BFGS-B).

Hybrid method :

Creation of QNEM, a mix of EM algorithm and a quasi-Newton type method (BFGS).

Introduction : Parameter estimation

Classical optimisation methods to estimate HMM parameters θ

Expectation-Maximisation (EM) algorithm :

Dempster et al., 1977; Baum and Petrie, 1966; Welch, 2003

Estimation of parameters by a variant of EM algorithm : Baum-Welch algorithm.

Direct maximisation of likelihood (DML) :

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

Likelihood maximisation by a quasi-Newton type method (L-BFGS-B).

Hybrid method :

Creation of QNEM, a mix of EM algorithm and a quasi-Newton type method (BFGS).

EM acceleration :

Varadhan and Roland, 2008

SQUAREM method (squared iterative methods).

⇒ Comparison of the 4 methods

Direct maximisation of the likelihood

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

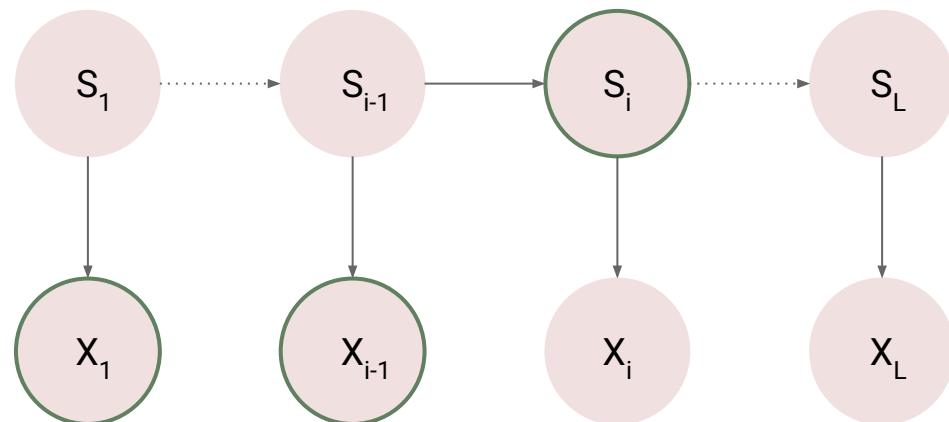
- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
 - ⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
 - direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - joint probabilities

Direct maximisation of the likelihood

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

$$a_i(s) = \mathbb{P}(S_i = s, X_1^{i-1} = x_1^{i-1})$$

- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
→ direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - joint probabilities



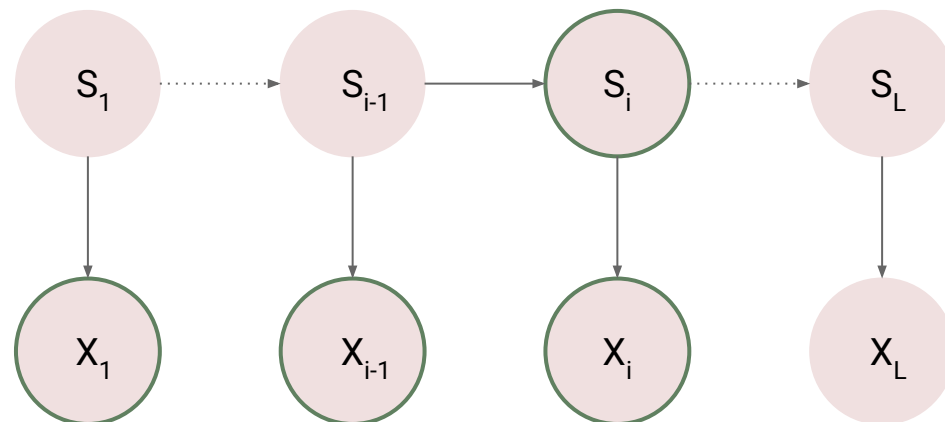
Direct maximisation of the likelihood

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

$$a_i(s) = \mathbb{P}(S_i = s, X_1^{i-1} = x_1^{i-1})$$

$$b_i(s) = \mathbb{P}_\theta(S_i = s, X_1^i = x_1^i)$$

- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
→ direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - joint probabilities



Direct maximisation of the likelihood

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
→ direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - joint probabilities

$$a_i(s) = \mathbb{P}(S_i = s, X_1^{i-1} = x_1^{i-1})$$

$$b_i(s) = \mathbb{P}_\theta(S_i = s, X_1^i = x_1^i)$$

1. Initialisation :

$$a_1(s) = \pi(s) \quad (1)$$

with $\pi(s)$ the initial distribution of state s .

2. Recursion:

$$\sum_t b_{i-1}(t) T_\theta(t, s) \quad (3)$$

$$= a_i(s) E_\theta(x_i, s) \quad (4)$$

where the sum in (3) is over t .

and can be computed as

$$= x_1^L) = \sum_s a_L(s) . \quad (5)$$

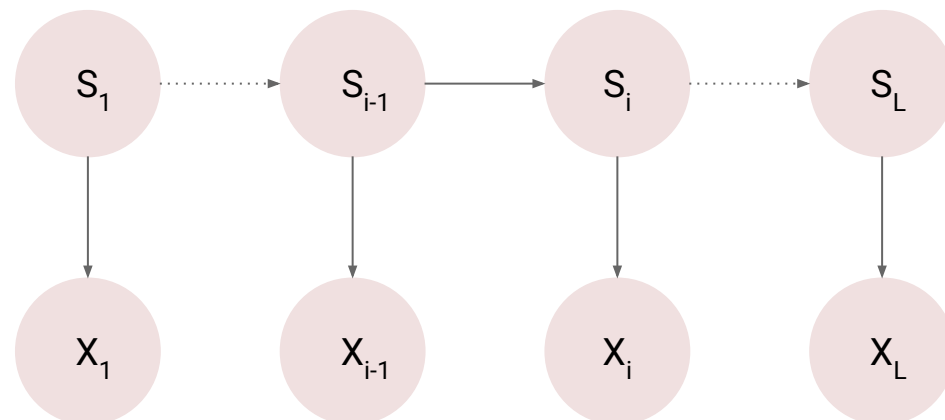
$$a_i(s) =$$

$$b_i(s) :$$

for $i \in 2, \dots, L$, w
all possible states

Finally, the likelihood

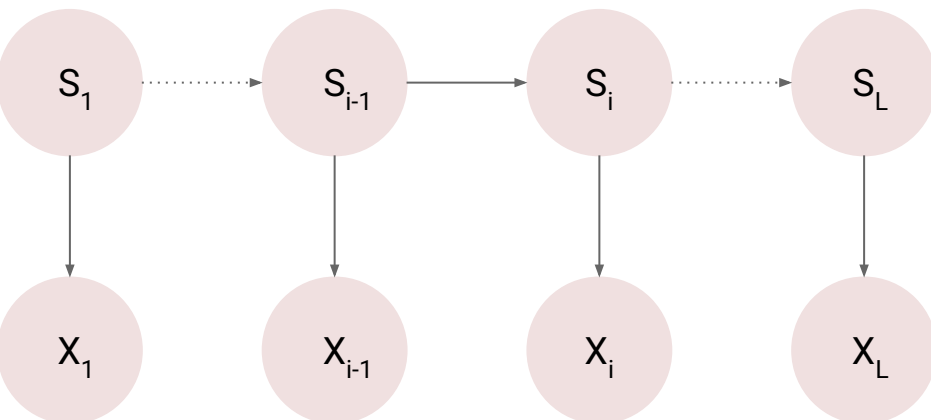
$$L(\theta; \mathbf{X}) = \mathbb{P}_\theta(X_1^L$$



Direct maximisation of the likelihood

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
→ direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - joint probabilities
 - need to compute everything on log scale



$$a_i(s) = \mathbb{P}(S_i = s, X_1^{i-1} = x_1^{i-1})$$

$$b_i(s) = \mathbb{P}_\theta(S_i = s, X_1^i = x_1^i)$$

1. Initialisation :

$$a_1(s) = \pi(s) \quad (1)$$

with $\pi(s)$ the initial distribution of state s .

2. Recursion:

$$\sum_t b_{i-1}(t) T_\theta(t, s) \quad (3)$$

$$= a_i(s) E_\theta(x_i, s) \quad (4)$$

where the sum in (3) is over t .

and can be computed as

$$= x_1^L) = \sum_s a_L(s) . \quad (5)$$

Finally, the likelihood

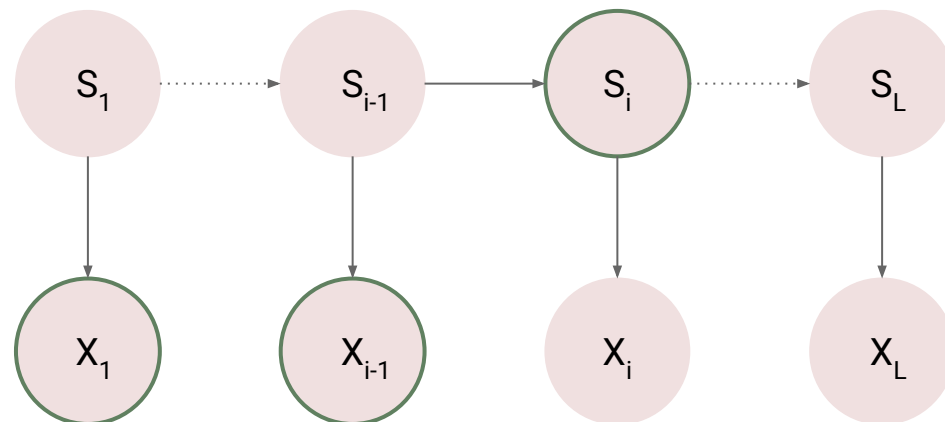
$$L(\theta; \mathbf{X}) = \mathbb{P}_\theta(X_1^L)$$

Direct maximisation of the likelihood

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

$$\alpha_i(s) = \mathbb{P}_\theta(S_i = s | X_1^{i-1} = x_1^{i-1})$$

- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
→ direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - ~~joint probabilities~~
 - ~~need to compute everything on log scale~~
 - **conditional probabilities**



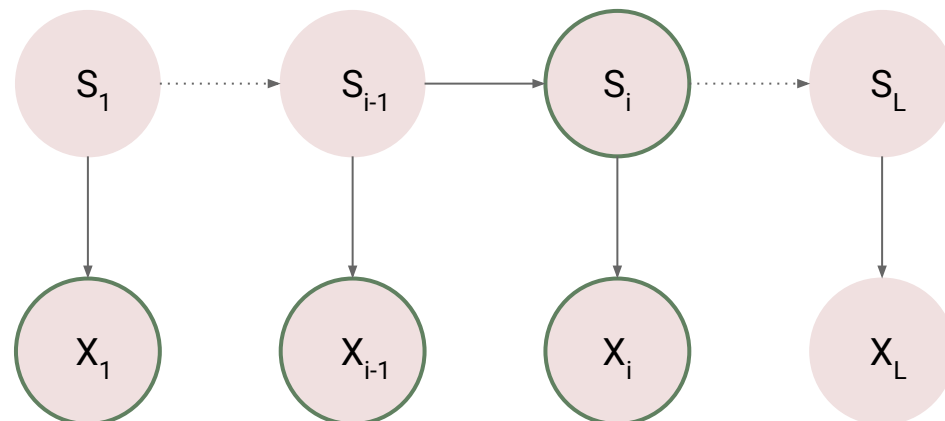
Direct maximisation of the likelihood

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

$$\alpha_i(s) = \mathbb{P}_\theta(S_i = s | X_1^{i-1} = x_1^{i-1})$$

$$\beta_i(s) = \mathbb{P}_\theta(S_i = s | X_1^i = x_1^i)$$

- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
→ direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - ~~joint probabilities~~
 - ~~need to compute everything on log scale~~
 - **conditional probabilities**



Direct maximisation of the likelihood

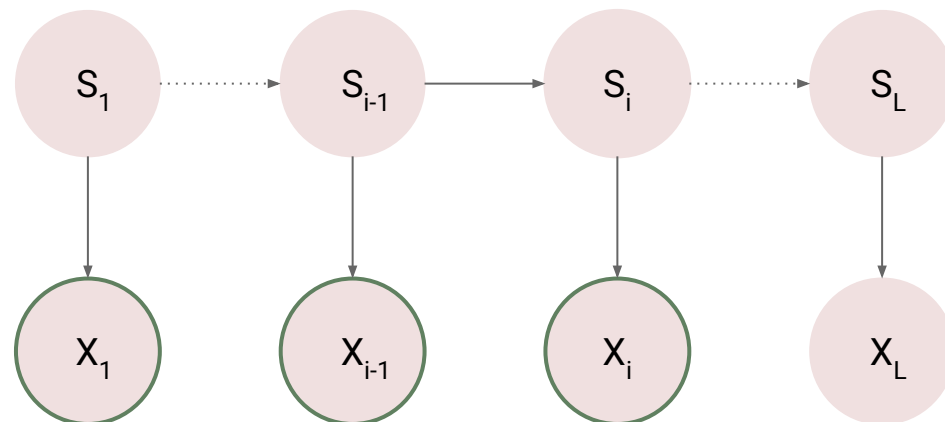
Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
→ direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - ~~joint probabilities~~
 - ~~need to compute everything on log scale~~
 - **conditional probabilities**

$$\alpha_i(s) = \mathbb{P}_\theta(S_i = s | X_1^{i-1} = x_1^{i-1})$$

$$\beta_i(s) = \mathbb{P}_\theta(S_i = s | X_1^i = x_1^i)$$

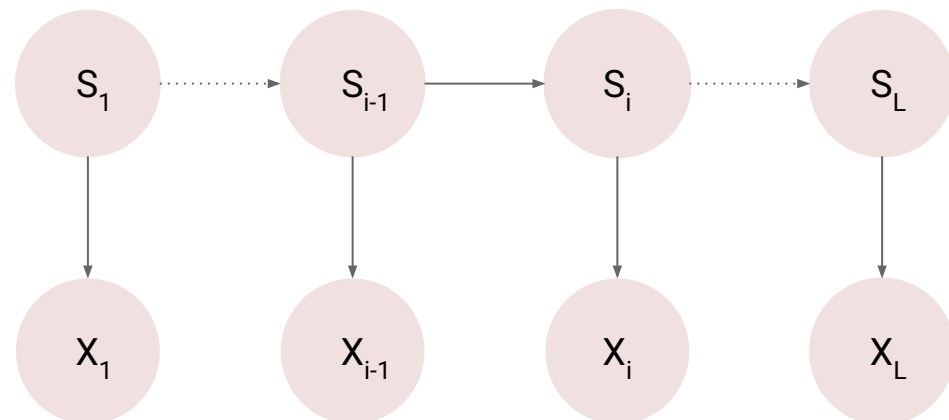
$$\gamma_i = \log(\mathbb{P}_\theta(X_1^i = x_1^i))$$



Direct maximisation of the likelihood

Broyden, 1970; Fletcher, 1970; Goldfarb, 1970; Shanno, 1970

- ❖ Quasi-Newton methods
 - maximisation of differentiable scalar functions
 - avoiding computing the Hessian matrix
⇒ approximating the inverse Hessian matrix at each iteration
 - method used : L-BFGS-B (Limited-memory Broyden-Fletcher-Goldfarb-Shanno with Bound constraints)
- ❖ Forward step → likelihood
→ direct maximisation with L-BFGS-B
- ❖ 2 possible forward algorithms :
 - ~~joint probabilities~~
 - ~~need to compute everything on log scale~~
 - **conditional probabilities**



$$\alpha_i(s) = \mathbb{P}_\theta(S_i = s | X_1^{i-1} = x_1^{i-1})$$

$$\beta_i(s) = \mathbb{P}_\theta(S_i = s | X_1^i = x_1^i)$$

$$\gamma_i = \log(\mathbb{P}_\theta(X_1^i = x_1^i)).$$

1. Initialisation:

$$\alpha_1(s) = \pi_s \quad (8)$$

$$\beta_1(s) = \frac{E_\theta(x_1, s) \times \alpha_1(s)}{\sum_i E_\theta(x_1, i) \times \alpha_1(i)} \quad (9)$$

$$\gamma_1 = \log \left(\sum_s \alpha_1(s) E_\theta(x_1, s) \right). \quad (10)$$

2. Recursion:

$$\alpha_i(s) = \sum_t T_\theta(t, s) \times \beta_{i-1}(t) \quad (11)$$

$$\beta_i(s) = \frac{E_\theta(x_i, s) \times \alpha_i(s)}{\sum_t E_\theta(x_i, t) \times \alpha_i(t)} \quad (12)$$

$$\gamma_i = \gamma_{i-1} + \log \left(\sum_s E_\theta(x_i, s) \alpha_i(s) \right) \quad (13)$$

for $i \in 2, \dots, L$

3. Log-likelihood : $\ell(\theta; \mathbf{X}) = \gamma_L$

EM algorithm (Baum-Welch)

Dempster et al., 1977; Baum and Petrie, 1966; Welch, 2003

E step : knowing θ , compute the probabilities of the hidden states along the chain given all the observations $\Rightarrow$ forward-backward algorithm

- ❖ **conditional** forward algorithm
- ❖ backward algorithm

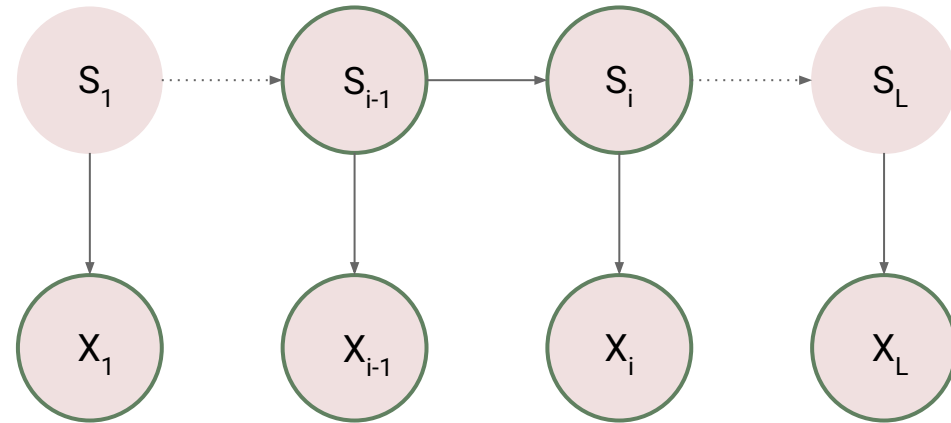
EM algorithm (Baum-Welch)

Dempster et al., 1977; Baum and Petrie, 1966; Welch, 2003

E step : knowing θ , compute the probabilities of the hidden states along the chain given all the observations $\Rightarrow$ forward-backward algorithm

- ❖ **conditional** forward algorithm
- ❖ backward algorithm :

$$\delta_i(s, t) = \mathbb{P}_{\theta}(S_{i-1} = s, S_i = t | X_1^L = x_1^L)$$



EM algorithm (Baum-Welch)

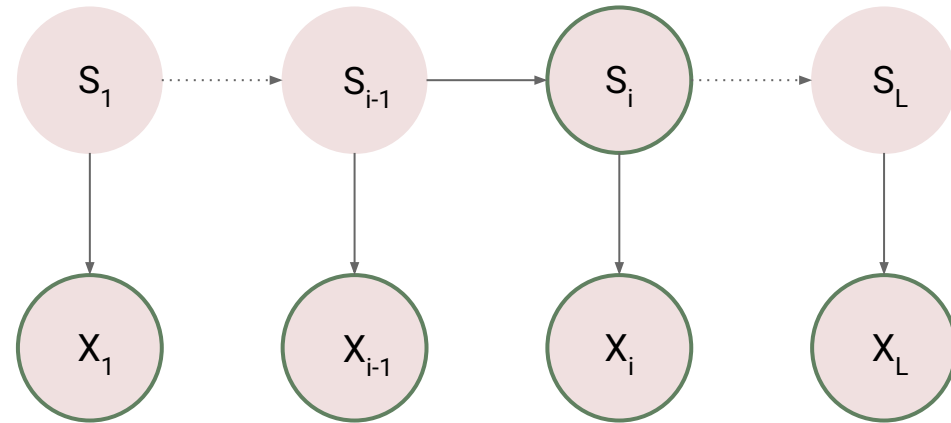
Dempster et al., 1977; Baum and Petrie, 1966; Welch, 2003

E step : knowing θ , compute the probabilities of the hidden states along the chain given all the observations $\Rightarrow$ forward-backward algorithm

- ❖ **conditional** forward algorithm
- ❖ backward algorithm :

$$\delta_i(s, t) = \mathbb{P}_{\theta}(S_{i-1} = s, S_i = t | X_1^L = x_1^L)$$

$$\varphi_i(s) = \mathbb{P}_{\theta}(S_i = s | X_1^L = x_1^L)$$



EM algorithm (Baum-Welch)

Dempster et al., 1977; Baum and Petrie, 1966; Welch, 2003

E step : knowing θ , compute the probabilities of the hidden states along the chain given all the observations $\Rightarrow$ forward-backward algorithm

❖ **conditional** forward algorithm

❖ backward algorithm :

$$\delta_i(s, t) = \mathbb{P}_{\theta}(S_{i-1} = s, S_i = t | X_1^L = x_1^L)$$

$$\varphi_i(s) = \mathbb{P}_{\theta}(S_i = s | X_1^L = x_1^L)$$

1. Initialisation:

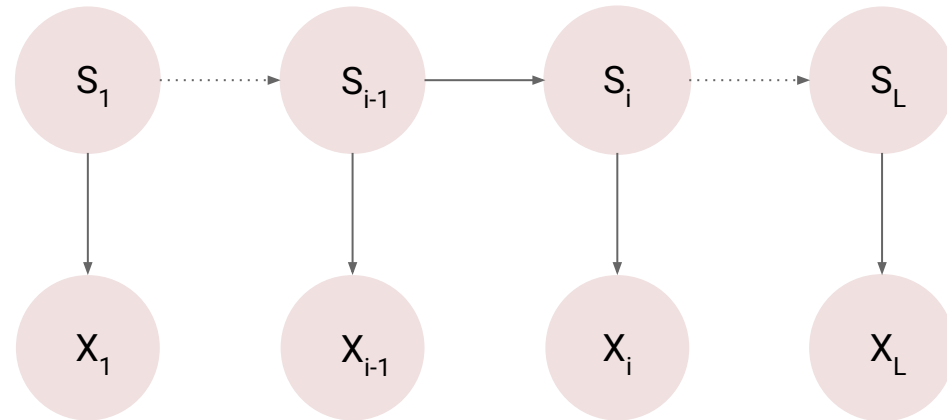
$$\varphi_L(s) = \beta_L(s) \quad (16)$$

2. Recursion:

$$\delta_i(s, t) = T_{\theta}(s, t) \times \frac{\beta_{i-1}(s)}{\alpha_i(t)} \times \varphi_i(t) \quad (17)$$

$$\varphi_{i-1}(s) = \sum_t \delta_i(s, t) \quad (18)$$

for $i \in L, \dots, 2$.



EM algorithm (Baum-Welch)

Dempster et al., 1977; Baum and Petrie, 1966; Welch, 2003

E step : knowing θ , compute the probabilities of the hidden states along the chain given all the observations $\Rightarrow$ forward-backward algorithm

- ❖ **conditional** forward algorithm
- ❖ backward algorithm :

$$\delta_i(s, t) = \mathbb{P}_{\theta}(S_{i-1} = s, S_i = t | X_1^L = x_1^L)$$

$$\varphi_i(s) = \mathbb{P}_{\theta}(S_i = s | X_1^L = x_1^L)$$

1. Initialisation:

$$\varphi_L(s) = \beta_L(s) \quad (16)$$

2. Recursion:

$$\delta_i(s, t) = T_{\theta}(s, t) \times \frac{\beta_{i-1}(s)}{\alpha_i(t)} \times \varphi_i(t) \quad (17)$$

$$\varphi_{i-1}(s) = \sum_t \delta_i(s, t) \quad (18)$$

for $i \in L, \dots, 2$.

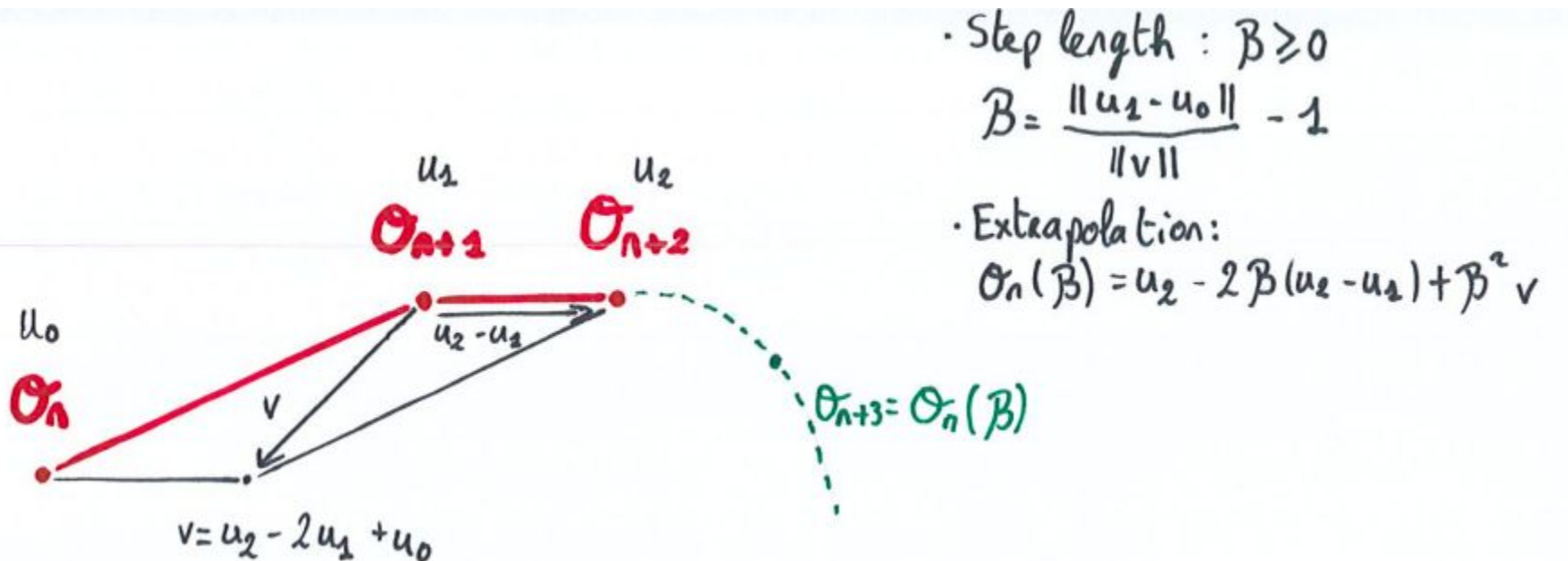
M step : re-estimate θ using these probabilities

- for each iteration of EM algorithm, estimation of the parameter θ for the next iteration
- if we knew the sequence of hidden state, we could compute θ with counting
- but the hidden states are unknown $\Rightarrow$ estimation of θ with probabilities

SQUAREM

Varadhan and Roland, 2008

- ❖ SQUAREM = Squared iterative Methods
- ❖ EM acceleration algorithm
- ❖ *Simple and Globally Convergent Methods for Accelerating the Convergence of Any EM Algorithm*, Varadhan et al., 2008, Scand. J. Stat.
- ❖ Aim : maintain the stability of EM while accelerating its convergence speed



Method : BFGS

Algorithm 2 BFGS algorithm

0. Input: $\theta^{(0)}$ = starting point.
Set $k = 0$ and $H_0 = \mathbb{I}$.
1. Let $p_k = -H_k \cdot \nabla f_k$ the search direction.
Perform a line search for α such that the Wolfe conditions are satisfied

$$f(\theta_k + \alpha \cdot p_k) \leq f(\theta_k) + c_1 \cdot \alpha (\nabla f_k)' p_k ;$$

$$\nabla f(\theta_k + \alpha \cdot p_k)' p_k \geq c_2 (\nabla f_k)' p_k .$$

2. Let $\theta^{(k+1)} = \theta^{(k)} + \alpha \cdot p_k$, $s_k = \theta^{(k+1)} - \theta^{(k)}$
and $y_k = \nabla f_{k+1} - \nabla f_k$.

3. BFGS (quasi-Newton) step: Update H_k using the BFGS formula

$$H_{k+1} = H_k - \frac{H_k \cdot y_k \cdot y_k' \cdot H_k}{y_k' \cdot H_k \cdot y_k} + \frac{s_k \cdot s_k'}{y_k' \cdot s_k} .$$

Let $k = k + 1$. Repeat 1. to 3. until convergence.

- ❖ Curvature condition :

$$s_k' \cdot y_k > 0$$

with $s_k = \theta_{k+1} - \theta_k$

and $y_k = \nabla f_{k+1} - \nabla f_k$

- ❖ With a complex line search, this condition is always met
- ❖ If the curvature condition is met, H_k stay positive definite

Proposed method : QNEM

Algorithm 1 QNEM algorithm

0. Input: $\theta^{(0)}$ = starting point.
Set $k = 0$ and $H_0 = \mathbb{I}$.

1. Baum-Welch (EM) step: Compute

$$\theta^{(k+1)} = BW(\theta^{(k)}) .$$

2. Let $s_k = \theta^{(k+1)} - \theta^{(k)}$ and $y_k = \nabla f_{k+1} - \nabla f_k$.
Repeat 1 & 2 until the curvature condition $s'_k \cdot y_k > 0$ is satisfied, or the convergence criterion (Section 3.5) is met.

3. BFGS (quasi-Newton) step: Update H_k using the BFGS formula

$$H_{k+1} = H_k - \frac{H_k \cdot y_k \cdot y'_k \cdot H_k}{y'_k \cdot H_k \cdot y_k} + \frac{s_k \cdot s'_k}{y'_k \cdot s_k} .$$

Let $k = k + 1$.

4. Let $p_k = -H_k \cdot \nabla f_k$ the search direction.
Perform a backtracking search for α such that the Armijo condition is satisfied

$$f(\theta_k + \alpha \cdot p_k) \leq f(\theta_k) + c \cdot \alpha (\nabla f_k)' p_k .$$

5. Let $\theta^{(k+1)} = \theta^{(k)} + \alpha \cdot p_k$.
If the curvature condition is satisfied, repeat 3 to 5 until the convergence criterion is met.
Else, reinitialise $H_k = \mathbb{I}$ and go back to 1.

- ❖ Simple line search = backtracking (Armijo condition)
- ❖ If the curvature condition is not met $\Rightarrow$ switch to EM

Example overview

Umbrella example :

Toy example : weather forecast
based on the presence of an
umbrella

- 2 hidden states
- discrete observations
(2 values)
- $N = 56$

Geyser example :

Example based on open access
data “Old Faithful Geyser” :
chain on the duration of geyser
eruptions
→ $N = 272$

2 sub-examples :

- dichotomised times :
 - 3 hidden states
 - discrete observations
(2 values)
- continuous times :
 - 3 hidden states
 - continuous observations

Genetics example :

Motivating problem for this
work (part of my PhD) :
identification of homozygous
segments. Data simulated from
1006 european HGDP-CEPH
(Human Genome Diversity
Project) haplotypes.

- 2 hidden states
- discrete observations
(3 values)
- $N = 1050$

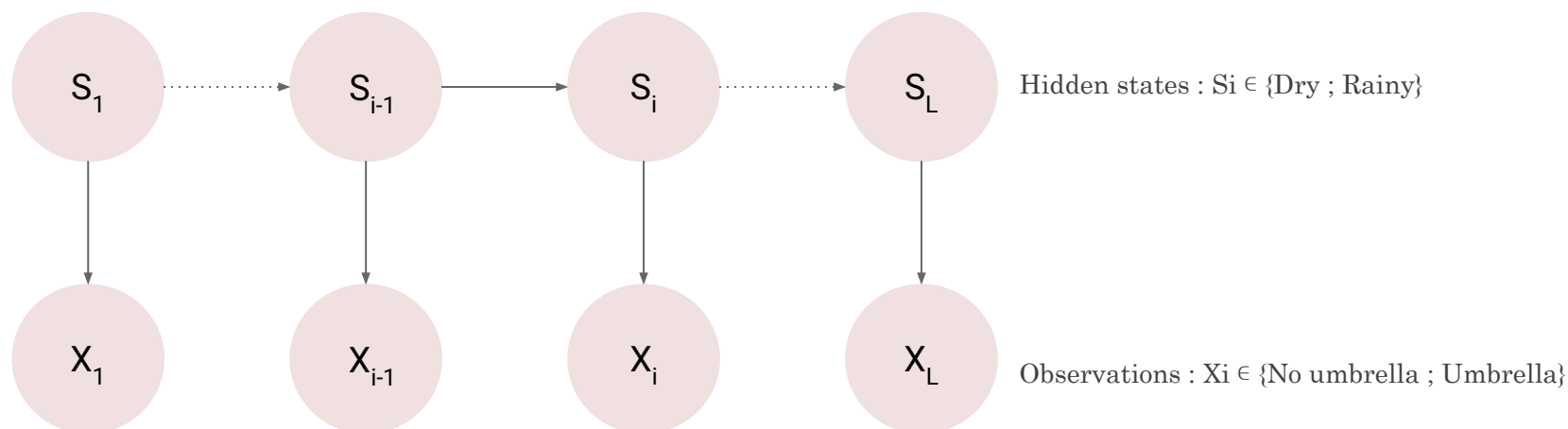
Umbrella example :

Toy example : weather forecast
based on the presence of an
umbrella

- 2 hidden states
- discrete observations
(2 values)
- $N = 56$

Transition matrix	$S_i = D$	$S_i = R$
$S_{i-1} = D$	$1-a$	a
$S_{i-1} = R$	a	$1-a$

Emission matrix	$S_i = D$	$S_i = R$
$X_i = N$	$1-b$	b
$X_i = U$	b	$1-b$



Results : Umbrella example

Umbrella example	Nb of iterations						Mean nb of steps		Mean time (s)	Percent of convergence
	Min	Q1	Q2	Mean	Q3	Max	Fw	Bw		
Quasi-Newton	4	13	16	15.77	18	37	15.77	0	0.03	100
Baum-Welch	3	18	24	27.95	31	338	27.95	27.95	0.06	100
SQUAREM	3	20	24	26.76	29	179	17.70	17.67	0.04	100
QNEM	2	8	11	11.32	13	118	12.63	2.80	0.03	100

The mean running times are in seconds.

Backward : ~1.7ms

Forward Baum-Welch & SQUAREM : ~0.5ms

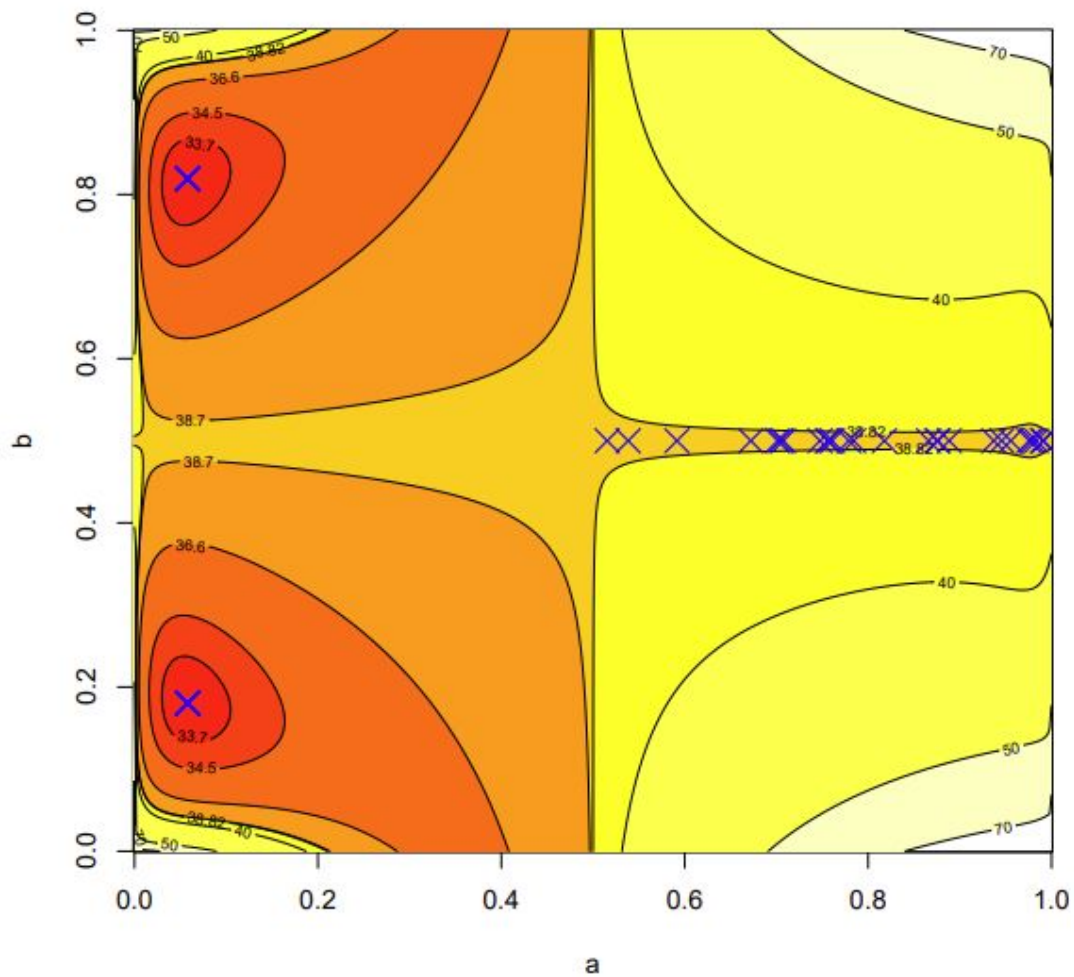
Forward quasi-Newton & QNEM : ~1.1ms

Results : Umbrella example

Likelihood	Parameters		Percent of permutations			
	a	b	Quasi-Newton	Baum-Welch	SQUAREM	QNEM
33.3	0.06	0.18 0.82	89.9	70.1	69.9	70.1
33.8	ND	0.5	10.1	29.9	30.1	29.3
Other			0	0	0	0.6

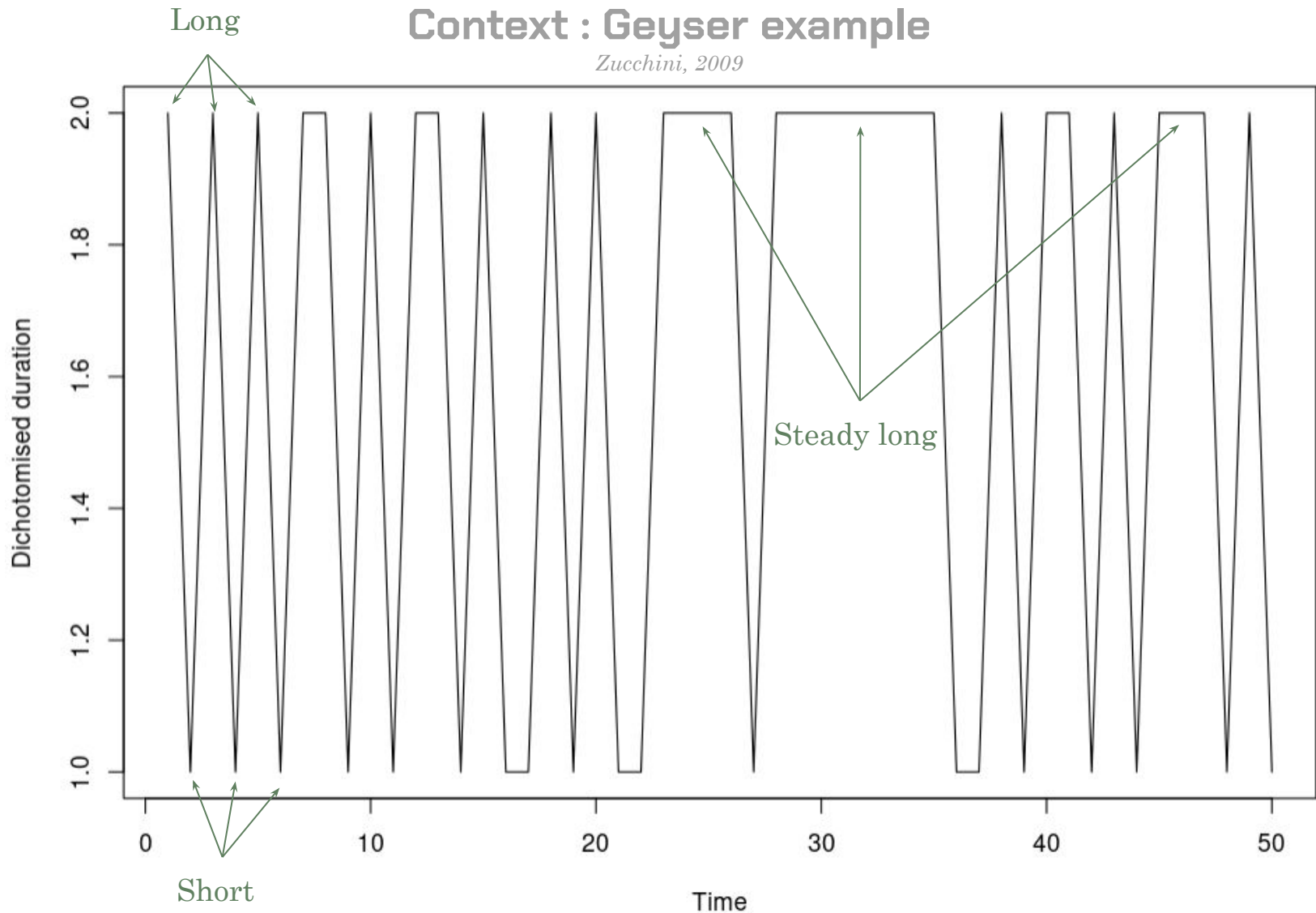
Table 8: This table shows the different convergence points reached with the umbrella example and their percent. ND = Non determined.

Log-likelihood with 100 convergence points



Context : Geyser example

Zucchini, 2009



Plot of dichotomised geyser eruptions' durations along time.

“1” code for eruption shorter than 3 minutes.

“2” code for eruption longer than 3 minutes.

Geyser example :

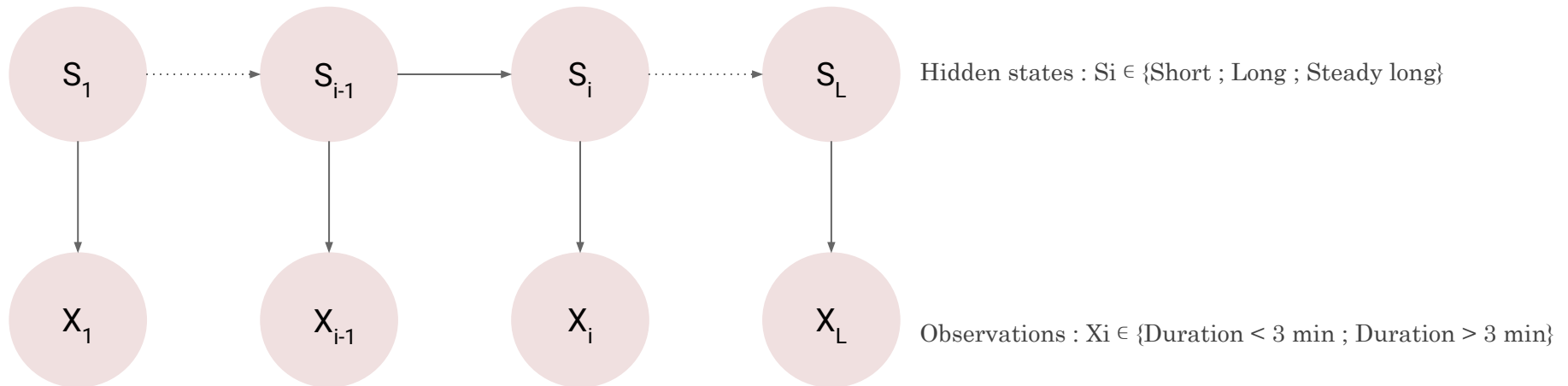
Example based on open access data “Old Faithful Geyser” :
chain on the duration of geyser eruptions
→ $N = 272$

2 sub-examples :

- **dichotomised** times :
→ 3 hidden states
→ discrete observations (2 values)
- continuous times :
→ 3 hidden states
→ continuous observations

Transition matrix	$S_i = S$	$S_i = L$	$S_i = Sl$
$S_{i-1} = S$	0	1-a	a
$S_{i-1} = L$	1	0	0
$S_{i-1} = Sl$	1-b	0	b

Emission matrix	$S_i = S$	$S_i = L$	$S_i = Sl$
$X_i = \text{Dinf3}$	1-c	1-d	1-e
$X_i = \text{Dsup3}$	c	d	e



Results : Dichotomised geyser example

Geyser example (dichotomised)	Nb of iterations						Mean nb of steps		Mean time (s)	Percent of convergence
	Min	Q1	Q2	Mean	Q3	Max	Fw	Bw		
Quasi-Newton	12	18	21	24.23	26	66	24.23	0	0.33	100
Baum-Welch	12	82	139	126	161	324	126	126	1.21	100
SQUAREM	15	71	113	108	140	290	78.59	71.89	0.71	100
QNEM	6	14	16	16.27	19	64	17.55	1.62	0.26	99.8

The mean running times are in seconds.

Backward : ~8ms

Forward Baum-Welch & SQUAREM : ~1.5ms

Forward quasi-Newton & QNEM : ~11.5ms

Results : Dichotomised geyser example

Likelihood	Parameters					Percent of permutations			
	a	b	c	d	e	Quasi-Newton	Baum-Welch	SQUAREM	QNEM
144.5	0.79	0.57	0*	1*	0.95	52.6	50.6	50.8	42.6
149.5	0.58	0.50	1*	0*	0.57	41.2	42.4	42.4	36.2
172.8	0* ND	ND 0*	0.55 / 0.74	0.74 / 0.55 ND	ND	6.2	7.0	0.9 5.9	4.4 4.5
Other						0	0	0	11.1

Table 10: This table shows the different convergence points reached with the dichotomised geyser example and their percent. ND = Non determined. * = These values were rounded for Quasi-Newton results because of the bound-constraints.

Geyser example :

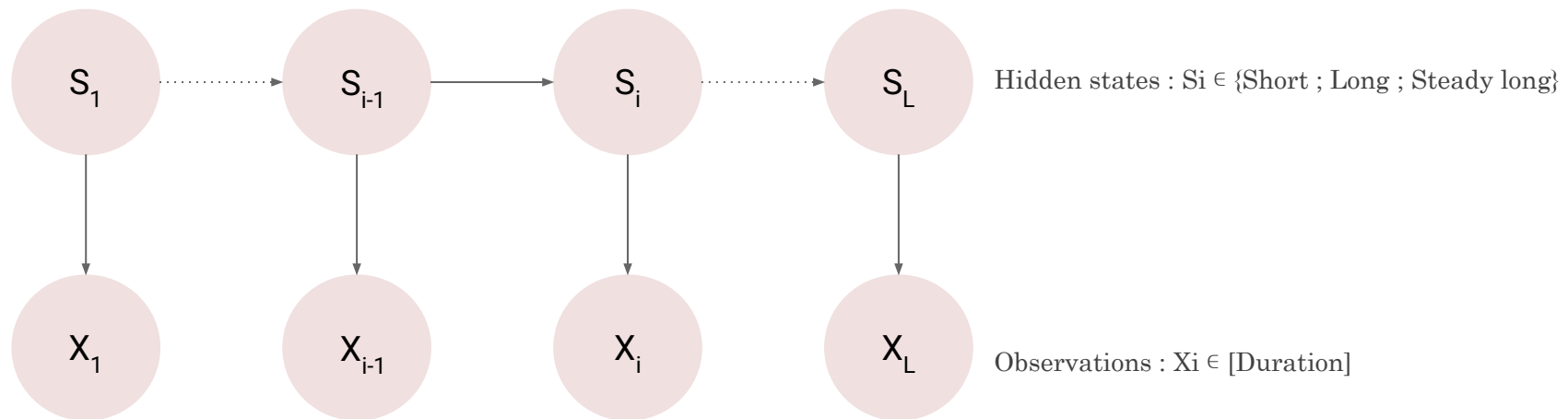
Example based on open access data “Old Faithful Geyser” :
chain on the duration of geyser eruptions
→ $N = 272$

2 sub-examples :

- dichotomised times :
→ 3 hidden states
→ discrete observations
(2 values)
- **continuous** times :
→ 3 hidden states
→ continuous observations

Transition matrix	$S_i = S$	$S_i = L$	$S_i = Sl$
$S_{i-1} = S$	0	1-a	a
$S_{i-1} = L$	1	0	0
$S_{i-1} = Sl$	1-b	0	b

The emission densities are the Gaussian densities with parameters μ_x and σ_x for state x .



Results : Continuous geyser example

Geyser example (continuous)	Nb of iterations						Mean nb of steps		Mean time (s)	Percent of convergence
	Min	Q1	Q2	Mean	Q3	Max	Fw	Bw		
Quasi-Newton	21	60	78	79.41	98	159	79.41	0	1.47	93.2
Baum-Welch	8	21	23	25.32	26	129	25.32	25.32	0.24	100
SQUAREM	12	30	32	34.04	35	118	23.23	22.59	0.22	100
QNEM	4	19	23	24.01	27	87	31.61	1.89	0.66	99.5

The mean running times are in seconds.

Backward : ~8ms

Forward Baum-Welch & SQUAREM : ~1.5ms

Forward quasi-Newton & QNEM : ~17.4ms

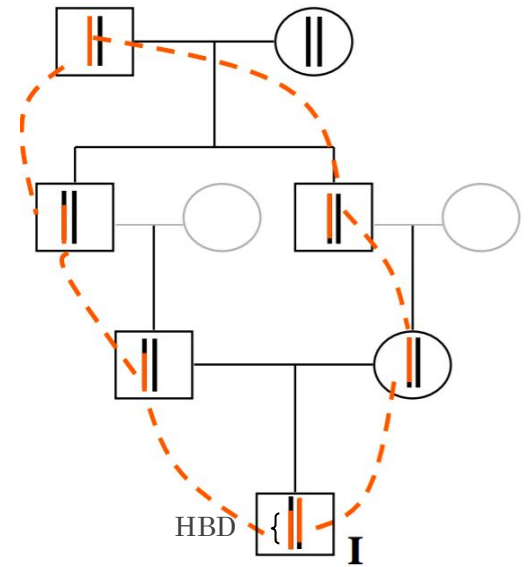
Results : Continuous geyser example

Likelihood	Parameters								Percent of permutations			
	a	b	μ_s	μ_l	μ_{sl}	σ_s	σ_l	σ_{sl}	Quasi-Newton	Baum-Welch	SQUAREM	QNEM
265.7	0.61	0.65	2.0	4.58	4.09	0.22	0.24	0.64	19.1	40.7	40.5	35.7
275.1	0.54	0.46	4.44	1.98	3.23	0.3	0.19	1.03	29.2	38.6	38.6	30.6
303.2	0.93	0.34	2.37	1.91	4.34	0.77	0.20	0.37	21.9	9.0	9.6	9.9
316.2	1*	0.28	4.45	ND	2.79	0.30	ND	1	0.1	0	0	6.4
335	0.44	0*	3.72	4.24	2.0	1.05	0.45	0.22	7.5	5.5	4.9	5.2
	0.56			2.0	4.24		0.22	0.45				
Other									22.2	6.2	6.4	12.2

Table 12: This table shows the different convergence points reached with the continuous geyser example and their percent. ND = Non determined. * = These values were rounded for Quasi-Newton results because of the bound-constraints.

Context : Genetics example

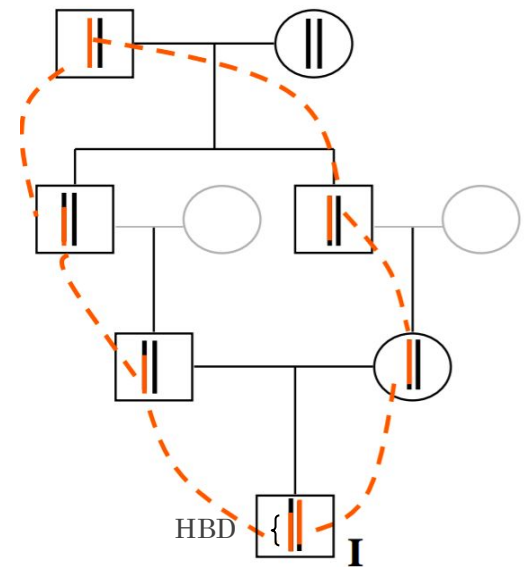
- ❖ Identification of rare recessive variants involved in multifactorial disease
- ❖ Analysis on consanguineous individuals (offspring of relatives), which are more likely to carry this type of variants
- ❖ **Inbreeding coefficient f** : probability that 2 alleles at a locus drawn in an individual's genome are identical and come from a common ancestor to his parents



Genealogy of an offspring of first cousins

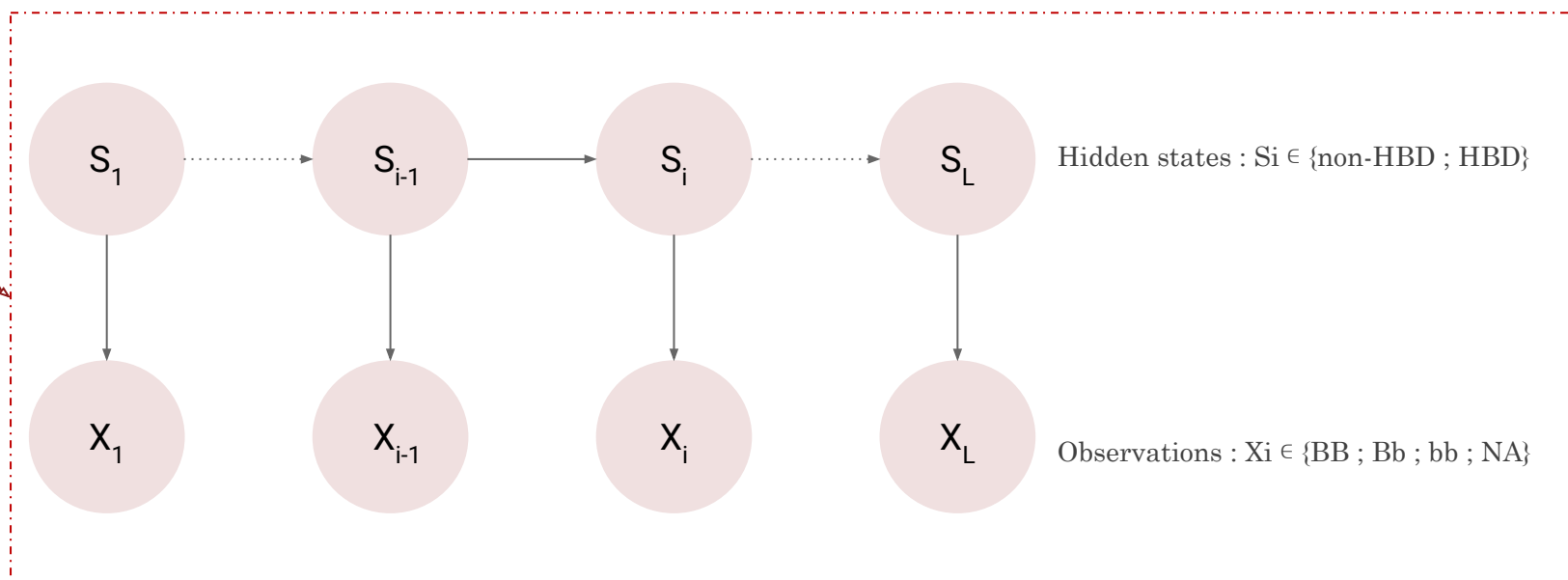
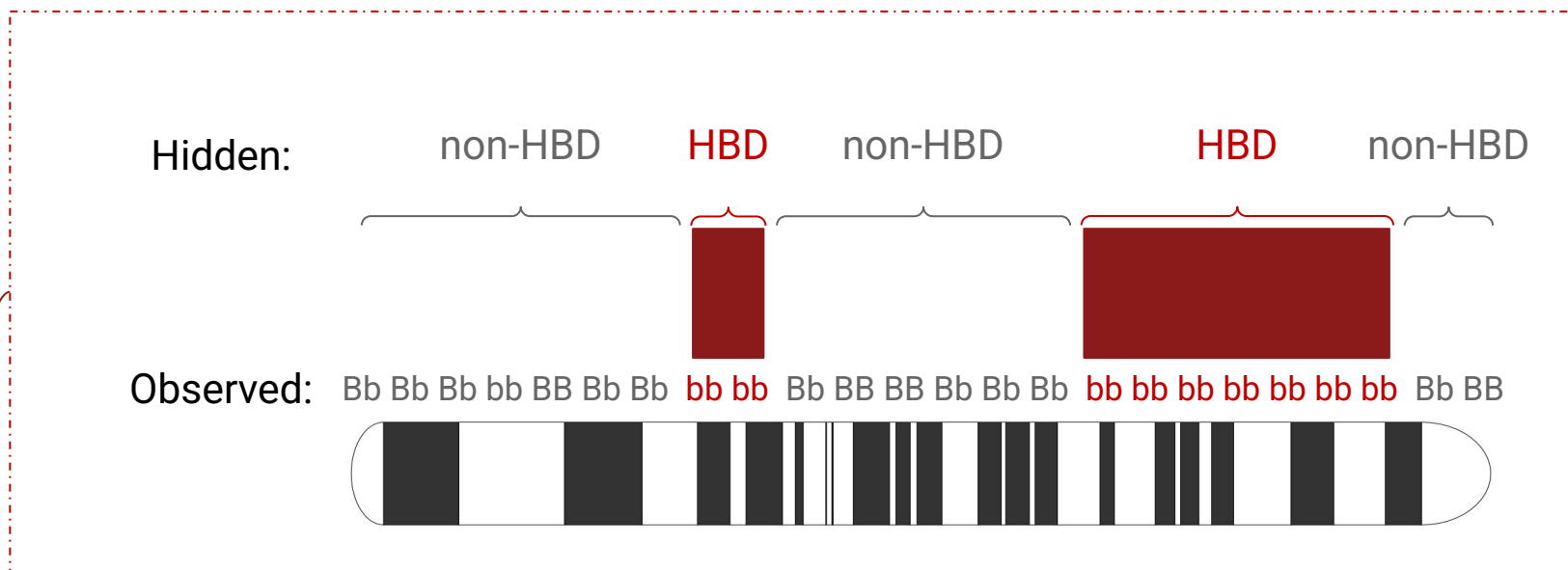
Context : Genetics example

- ❖ In absence of family data : identified through his genome which carries long homozygous segments : **HBD segments**
- ❖ We introduce α as : the mean length of HBD segments is $1/(\alpha(1-f))$ and the mean length of non-HBD segments is $1/(\alpha f)$ (in cM)
- ❖ HBD segments inference $\Rightarrow$ Hidden Markov model (**HMM**)



Genealogy of an offspring of first cousins

HMM



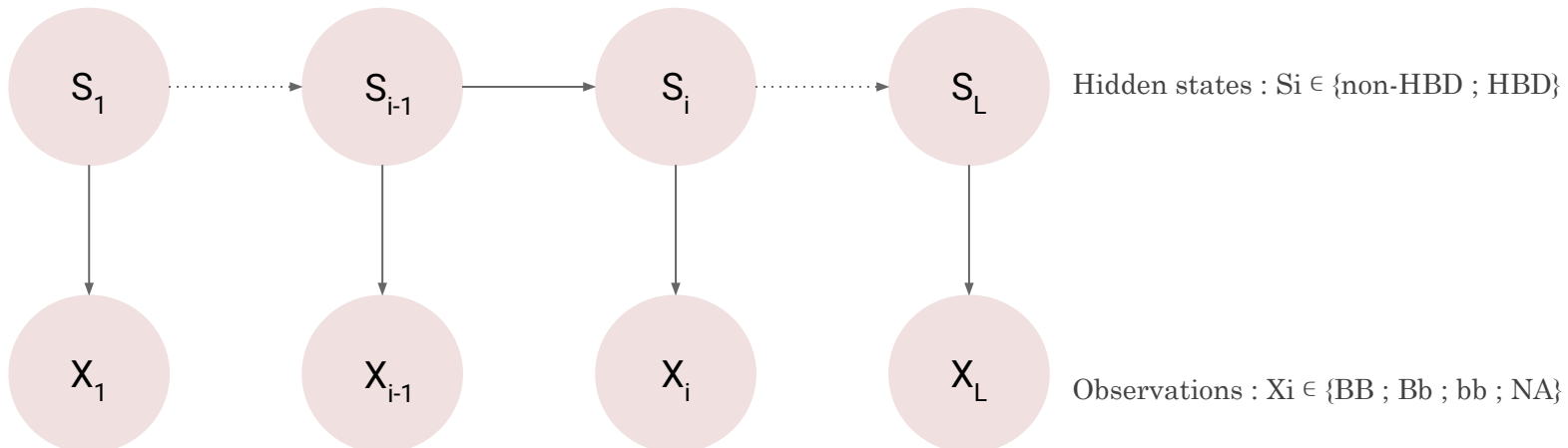
Genetics example :

Motivating problem for this work (part of my PhD) :
identification of homozygous segments. Data simulated from 1006 european HGDP-CEPH (Human Genome Diversity Project) haplotypes.

- 2 hidden states
- discrete observations
(3 values)
- $N = 1050$

Transition matrix	$S_i = \text{nHBD}$	$S_i = \text{HBD}$
$S_{i-1} = \text{nHBD}$	$(1 - \exp^{-ad})(1 - f) + \exp^{-ad}$	$(1 - \exp^{-ad})f$
$S_{i-1} = \text{HBD}$	$(1 - \exp^{-ad})(1 - f)$	$(1 - \exp^{-ad})f + \exp^{-ad}$

Emission matrix	$S_i = \text{nHBD}$	$S_i = \text{HBD}$
$X_i = \text{BB}$	p_B^2	$(1 - \epsilon)p_B + \epsilon \cdot p_B^2$
$X_i = \text{Bb}$	$2 \cdot p_B \cdot p_b$	$2\epsilon p_B \cdot p_b$
$X_i = \text{bb}$	p_b^2	$(1 - \epsilon)p_b + \epsilon \cdot p_b^2$
$X_i = \text{NA}$	1	1



Results : HBD segments example

HBD segments example	Nb of iterations						Mean nb of steps		Mean time (s)	Percent of convergence
	Min	Q1	Q2	Mean	Q3	Max	Fw	Bw		
Quasi-Newton	7	11	14	15.36	18	38	15.36	0	0.32	100
Baum-Welch	5	63	73	68.53	78	86	68.53	68.53	2.40	100
SQUAREM	5	50	57	56.19	63	88	38.66	38.66	1.37	100
QNEM	5	9	11	11.99	14	27	12.83	1.40	0.32	100

The mean running times are in seconds.

Backward : ~30.3ms

Forward Baum-Welch & SQUAREM : ~5.3ms

Forward quasi-Newton & QNEM : ~20.2ms

Conclusion & discussion

- Proposed QNEM $\Rightarrow$ take advantage of Expectation-Maximisation algorithm and Direct maximisation of the likelihood
 - Expectation-Maximisation algorithm : Baum-Welch algorithm
 - + stay close to the solution
 - - takes time to converge
 - Direct maximisation of the likelihood : quasi-Newton algorithm
 - + converge faster
 - - needs proper initialisation
- Evaluated SQUAREM to measure the extent of the acceleration
- QNEM showed the best results in all example except continuous geyser example where EM / SQUAREM are best
- For genetics example quasi-Newton algorithm and QNEM are equivalent
- No uniformly better algorithm
 - speed of convergence
 - global vs. local maximum

QNEM availability

Submitted article (under revision) to
*Journal of Computational and Applied
Mathematics*

⇒ preprint available in *HAL* :
<https://hal.science/hal-04685772v2>

QNEM implemented in R

⇒ R package *steveHMM* available on
GitHub :

<https://github.com/SidonieFoulon/steveHMM>



To preprint



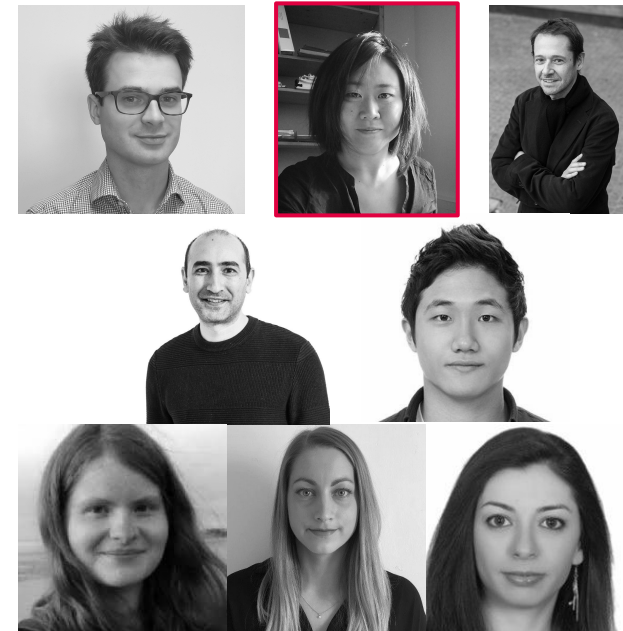
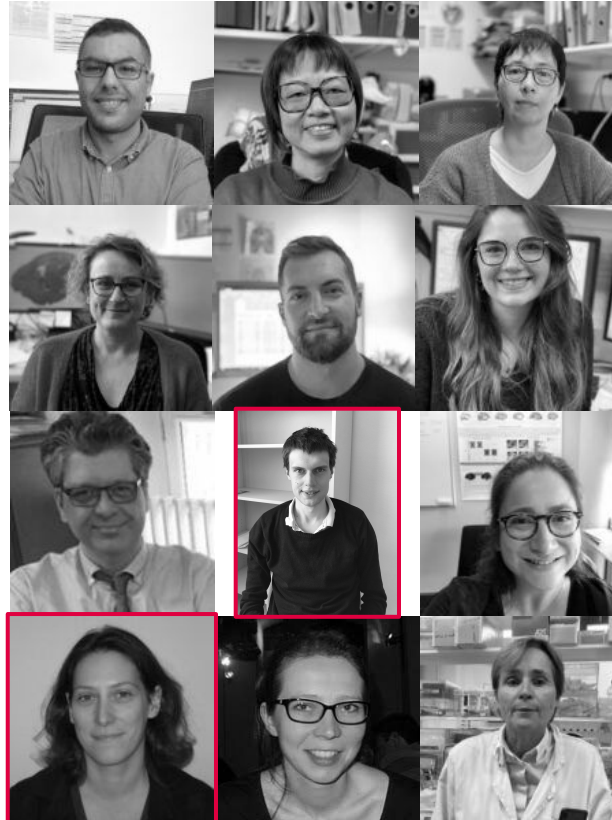
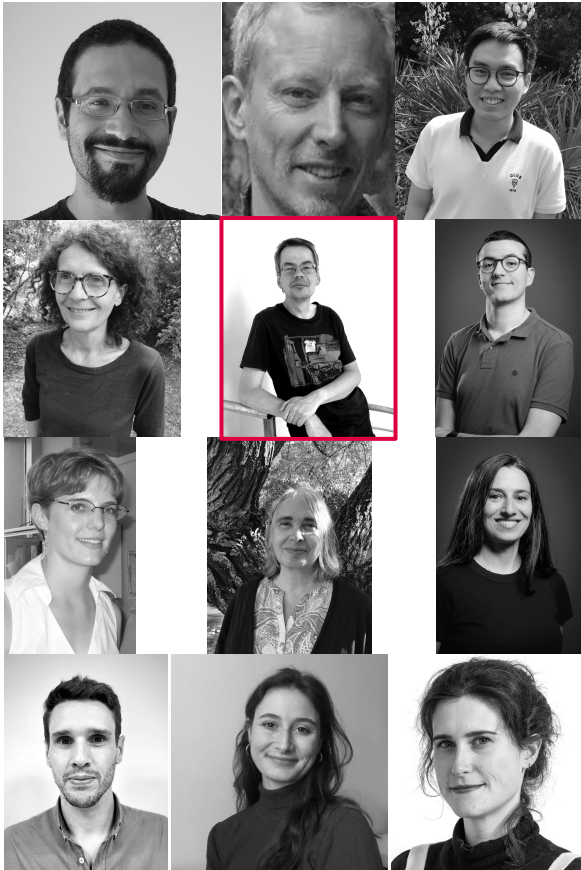
To R package

MERCI !

CESP HiDiBiostat

NeuroDiderot / ICM

CESP Exposome & heredity, genetic group



Supplementary

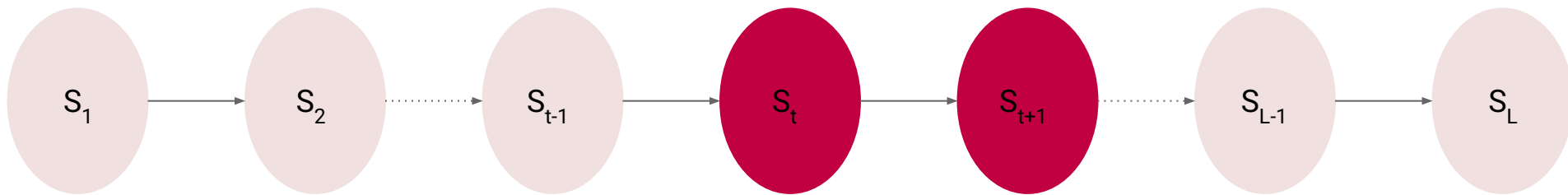
Introduction : From Markov models ...

A sequence of random variables measured at successive moments S_i is a **Markov chain** if it satisfies the Markov property :

- ★ to predict all the S_i subsequent to the time t , the information collected for $i \leq t$ is completely included in the single value S_t

$$\mathbb{P}(S_{t+1} | S_{1:t}) = \mathbb{P}(S_{t+1} | S_t)$$

- ★ S_i 's are not independent
- ★ S_i 's are independent conditionally on the previous observations



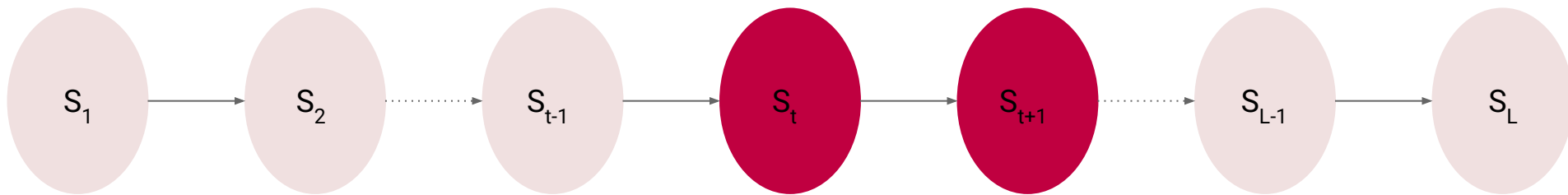
Introduction : From Markov models ...

Chain is **stationary** if : $\{S_{1+d}, \dots, S_{t+d}\}$ follow the same distribution as $\{S_1, \dots, S_t\}$

Homogeneity of the sequence :

- ❖ Markov chain converge fast to this stationary distribution
- ❖ **transition probabilities** $\mathbb{P}(S_{i+1}|S_i)$ are identical in every point of the sequence

➤ sometimes contradicted by the observations



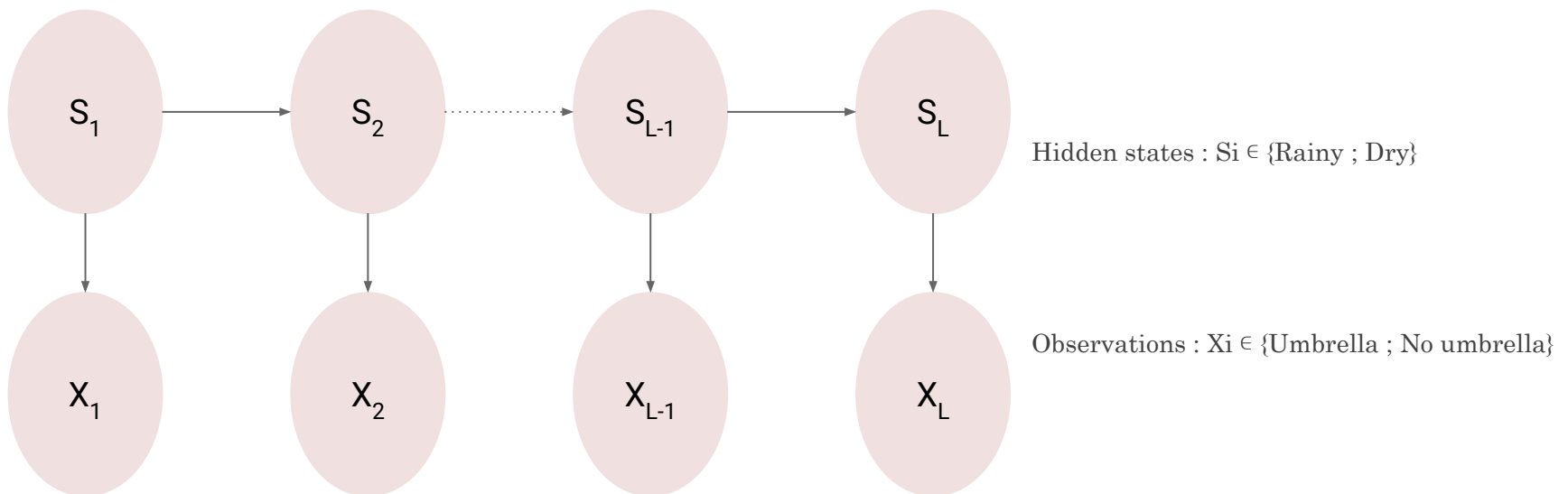
Introduction : ... to Hidden Markov models

To model such situations :

- ❖ add an **observable** layer of variables to the model
- ❖ related to the sequence of **hidden** variables

⇒ **Hidden Markov Models (HMM)**

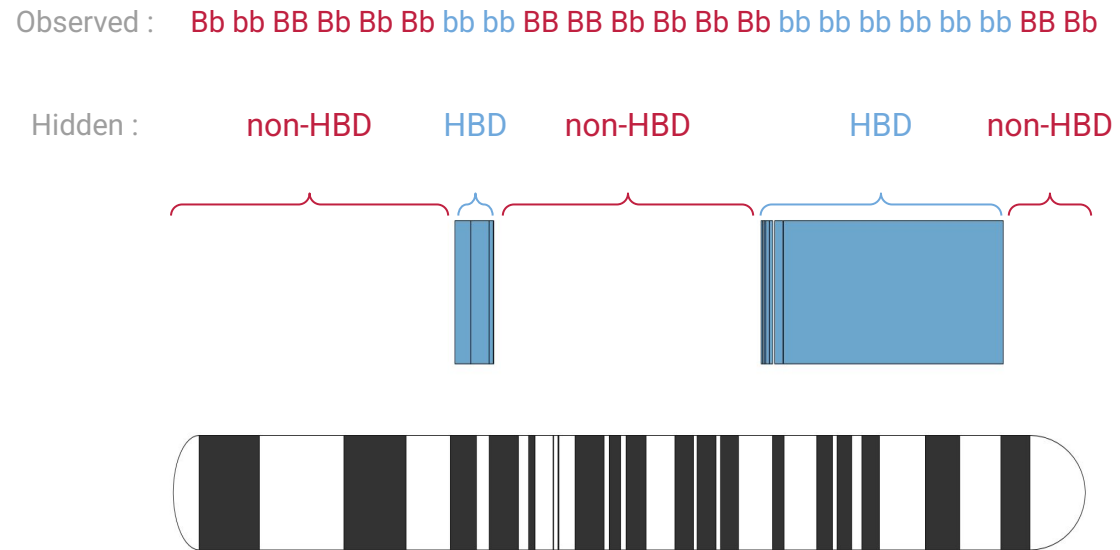
- ex : proportion of homozygous genotypes is almost 1 in HBD segments unlike the rest of the genome
- ◆ HMM to reconstruct an hidden information (HBD segments) from an observable information (genotypes)



Thesis context

- ❖ Identification of rare recessive variants involved in multifactorial disease
- ❖ Analysis on consanguineous individuals (offspring of relatives), which are more likely to carry this type of variants
- ❖ **Inbreeding coefficient f** : probability that 2 alleles at a locus drawn in an individual's genome are identical and come from a common ancestor to his parents
- ❖ In absence of family data : identified through his genome which carries long homozygous segments : **HBD segments**
- ❖ We introduce α as : the mean length of HBD segments is $1/(\alpha(1-f))$ and the mean length of non-HBD segments is $1/(\alpha f)$ (in cM)
- ❖ HBD segments inference $\Rightarrow$ Hidden Markov model (**HMM**)

Representation of the HBD segments in the chromosome 8 of a consanguineous individual



Methods : EM algorithm

E step : knowing θ , compute the probabilities of the hidden states along the chain knowing all the observations

⇒ forward-backward algorithm

- Forward initialize with $\alpha_1(s) = P_\theta(S_1 = s)$.
Then for each position $i \in \{2, \dots, L\}$, compute :
 - $\alpha_i(s) = P_\theta(S_i = s \mid X_1, \dots, X_{i-1})$
 - $\beta_i(s) = P_\theta(S_i = s \mid X_1, \dots, X_i)$
- Backward initialize with $\varphi_L(s) = \beta_L(s)$.
Then for each position $i \in \{L-1, \dots, 1\}$, compute :
 - $\delta_i(s, t) = P_\theta(S_{i+1} = s, S_i = t \mid \mathbf{X})$
 - $\varphi_i(s) = P_\theta(S_i = s \mid \mathbf{X})$

M step : re-estimate θ using these probabilities

The likelihood of θ is :

$$L(\theta ; \mathbf{S} = \mathbf{s}, \mathbf{X} = \mathbf{x}) = P_\theta(S_1 = s_1) \cdot P_\theta(S_2 = s_2 \mid S_1 = s_1) \cdot \dots \cdot P_\theta(S_L = s_L \mid S_{L-1} = s_{L-1}) \\ \cdot P_\theta(X_1 = x_1 \mid S_1 = s_1) \cdot \dots \cdot P_\theta(X_L = x_L \mid S_L = s_L)$$

$$\rightarrow l(\theta ; \mathbf{S} = \mathbf{s}, \mathbf{X} = \mathbf{x}) = \log P_\theta(S_1 = s_1) + \log P_\theta(S_2 = s_2 \mid S_1 = s_1) + \dots + \log P_\theta(S_L = s_L \mid S_{L-1} = s_{L-1}) \\ + \log P_\theta(X_1 = x_1 \mid S_1 = s_1) + \dots + \log P_\theta(X_L = x_L \mid S_L = s_L)$$

Expected value of the log-likelihood :

$$\rightarrow Q(\theta ; \theta^{(w)}) = E(l(\theta ; \mathbf{S} = \mathbf{s}, \mathbf{X} = \mathbf{x}) \mid \theta^{(w)}) \\ = \sum_s \log P_\theta(S_1 = s) \cdot \varphi_1(s) + \sum_{s,t} \log P_\theta(S_2 = t \mid S_1 = s) \cdot \delta_2(s, t) + \dots + \sum_{s,t} \log P_\theta(S_L = t \mid S_{L-1} = s) \cdot \delta_L(s, t) \\ + \sum_s \log P_\theta(X_1 = x_1 \mid S_1 = s) \cdot \varphi_1(s) + \dots + \sum_s \log P_\theta(X_L = x_L \mid S_L = s) \cdot \varphi_L(s) \quad \left. \vphantom{\sum_{s,t}} \right\} \text{to maximise}$$

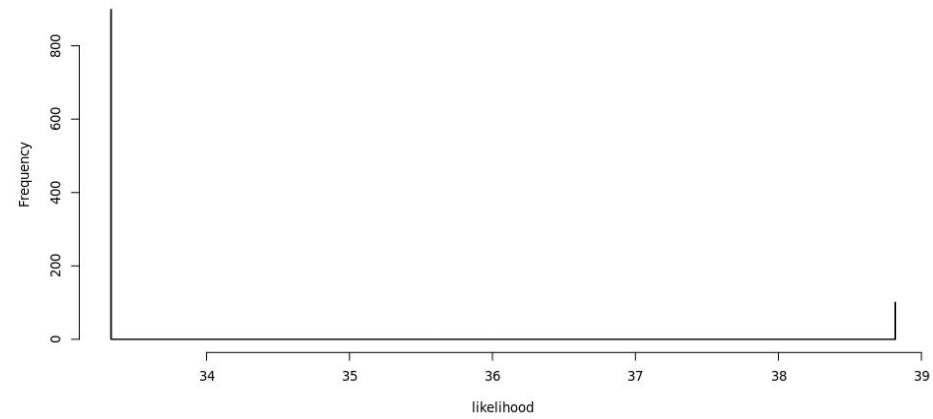
Pseudo code SQUAREM

Pseudocode :

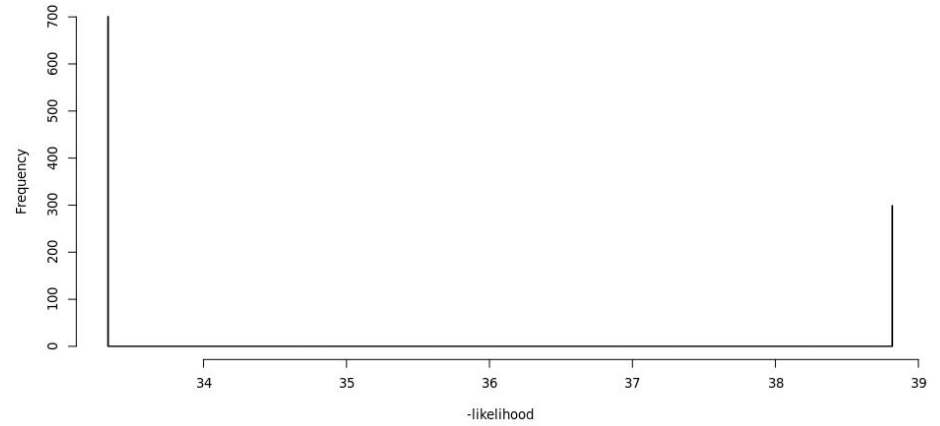
1. Initialise with θ_0
2. 2 EM iterations $\Rightarrow \theta_1$ et θ_2
3. $r = (\theta_0 - \theta_1)$
4. $v = (\theta_2 - \theta_1) - r$
5. Compute step length : norm ratio of r and v
6. Estimate θ' : with θ_0 , step length, r and v
7. Update θ_0 with $\text{EM}(\theta')$
8. Check convergence
 - a. go back to 1.
 - b. or stop

Umbrella example

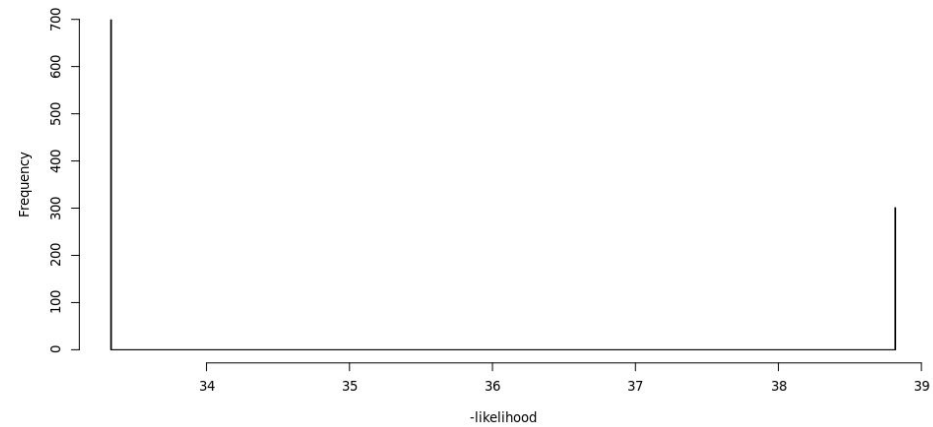
Histogram of likelihood, qN



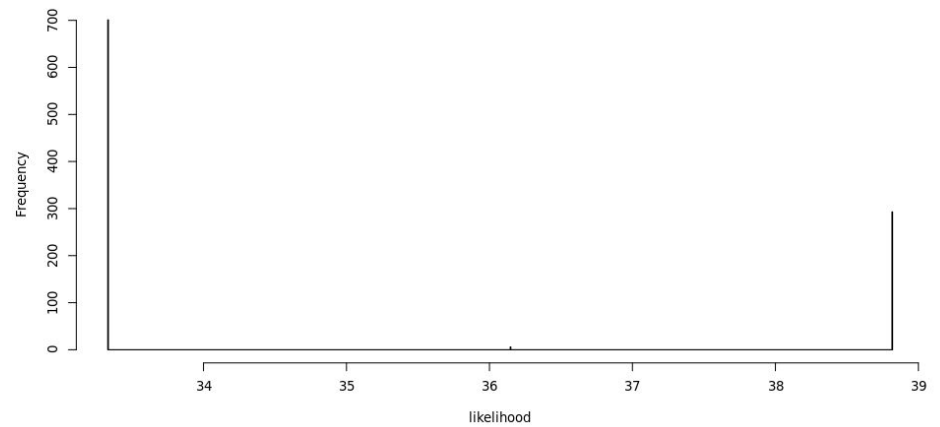
Histogram of likelihood, BW



Histogram of likelihood, SQUAREM

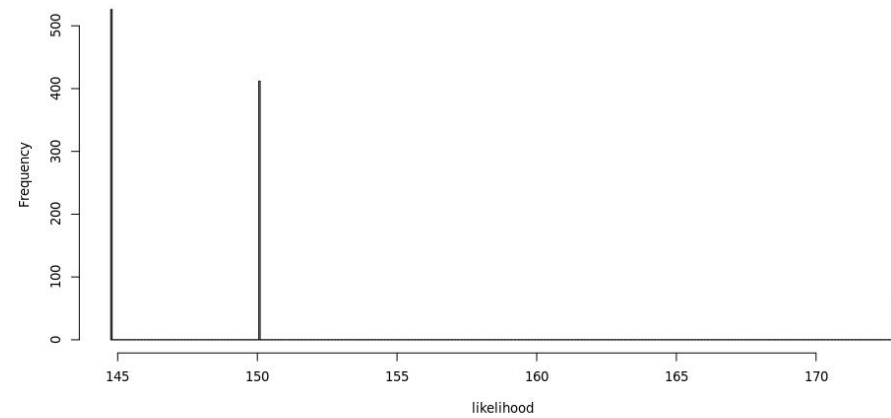


Histogram of likelihood, QNEM

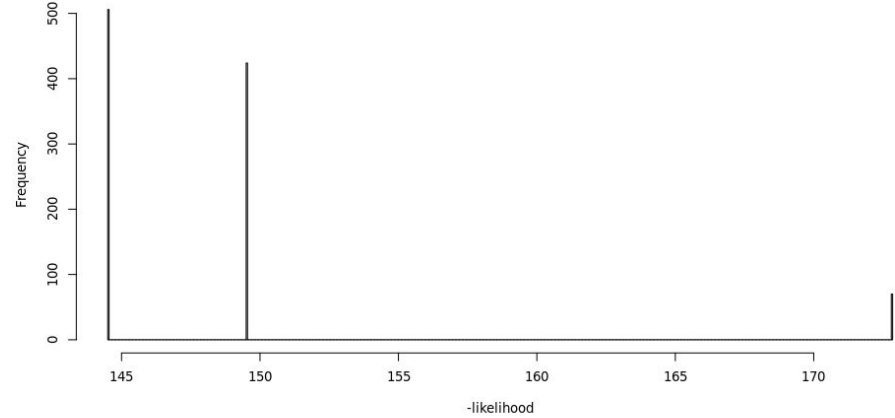


Dichotomised geyser example

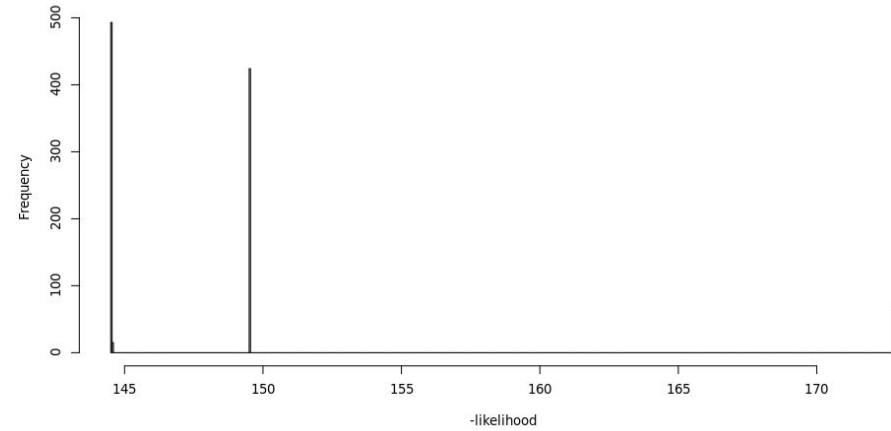
Histogram of likelihood, qN



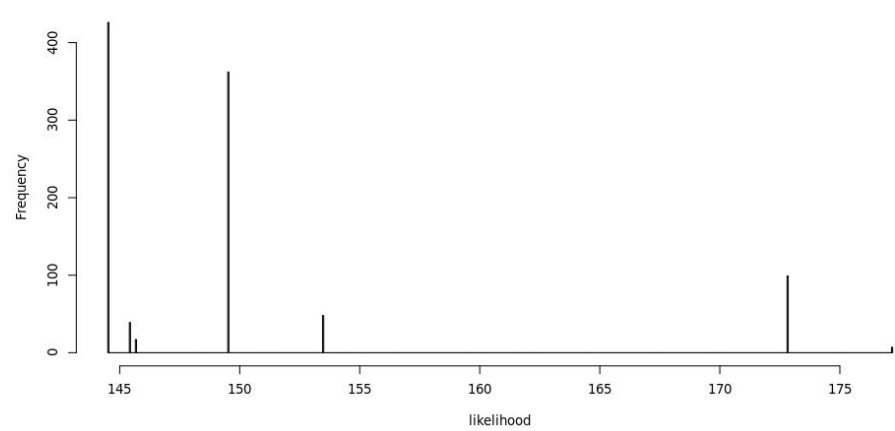
Histogram of likelihood, BW



Histogram of likelihood, SQUAREM

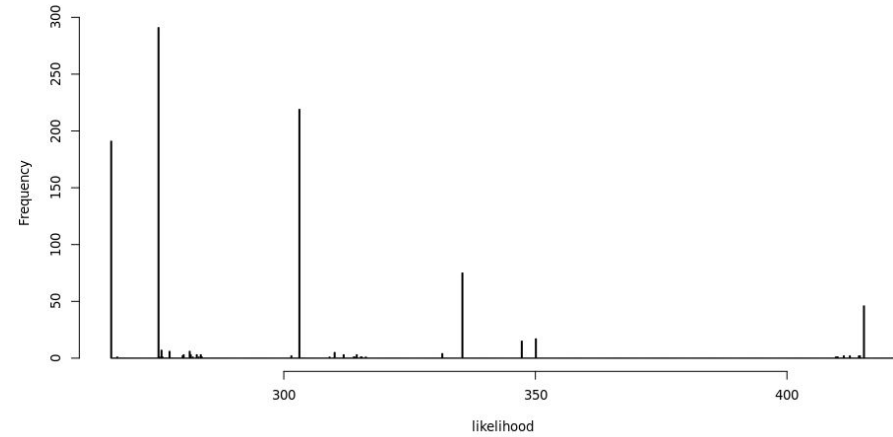


Histogram of likelihood, QNEM

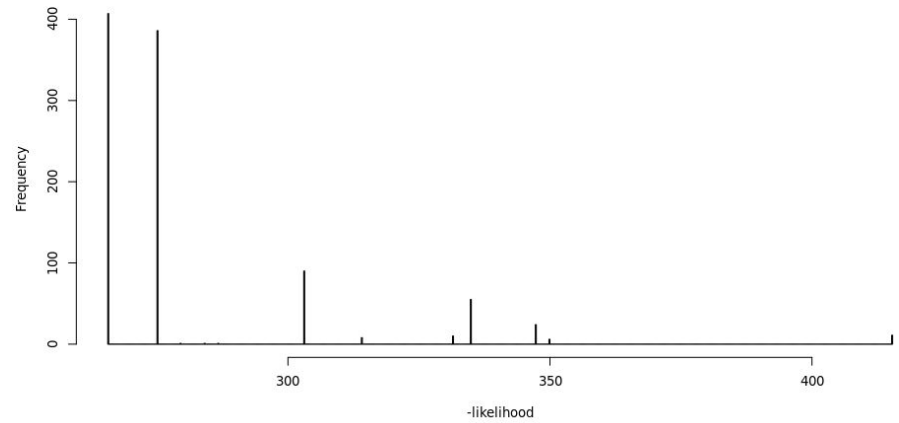


Continuous geyser example

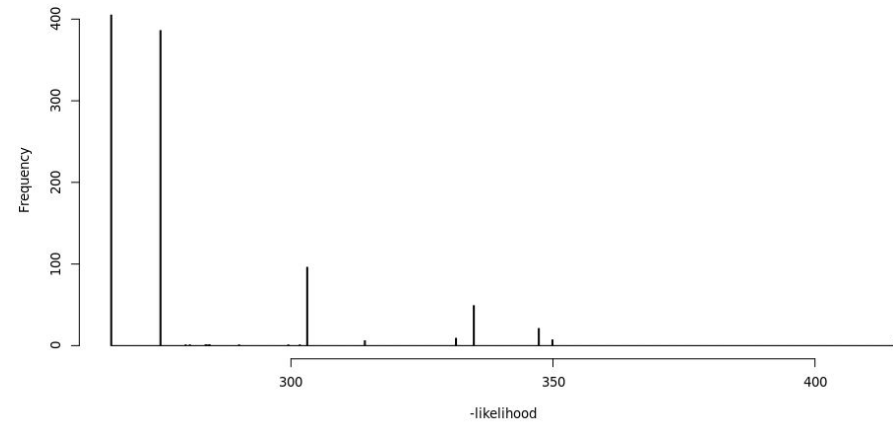
Histogram of likelihood, qN



Histogram of likelihood, BW



Histogram of likelihood, SQUAREM



Histogram of likelihood, QNEM

